

Sprint 4 Report

Team: eLation Nation

Project: eLation Mobile Apps

Project Sponsor: Innovative Systems LLC

Spring Duration: January 16, 2013 to February 6, 2013

Team members:

- Michael Malkowski
- Rachel Pekarek
- Jeremy Warner

Project Sponsor: Innovative Systems LLC

Meetings/Client Interactions

- 1/15/13
 - Sprint Planning Meeting
 - Determined Sprint 4 Goals
- 1/22/13
 - Scrum Meeting
 - Updated client with progress
- 1/29/13
 - Scrum Meeting
 - Updated client with progress
- 2/5/13
 - Scrum Meeting
 - Updated client with progress

Sprint Tasks

Our sprint tasks were divided into the following categories:

- Android
 - Technician Application
 - Set up saved Account functionality
 - Add a settings page
 - Redesign layout of Diagnostic data
 - Enable screen rotation
 - Enable horizontal scrolling
 - Complete App according to current requirements
 - Handling UI while loading data
 - eBill Application
 - Implement Sign In process
 - Redesign internal storage structure
 - Add database to project
 - Make Payments from phone
 - Retrieve real data for current usage
 - Read and edit real account settings data
 - Deploy both apps to Mitchell
- Apple
 - Technician Application
 - Set up saved Account functionality
 - Add a settings page
 - Make expandable diagnostics list
 - Enable screen rotation
 - Implement first time set up and save URL
 - Complete App according to current requirements
 - Enable Horizontal Scrolling
 - eBill Application
 - Implement Sign In process
 - Create internal storage structure
 - Add database to project
 - Make Payments from phone
 - Display real data for current usage
 - Read and edit real account settings data
- API
 - Technician Application
 - Investigate IMSI or Device-Specific Identifiers for Authentication
 - Implement Error Checking and reporting
 - Write Unit Tests
 - Parse Dates correctly
 - Add Logging functionality

- Add Call ID to calls
- eBill Application
 - Test making a new Payment
 - Write Unit Tests
 - Parse Dates correctly
 - Add Logging functionality
 - Write Trouble Ticket interface
 - Write Account List interface
 - Write Email Get/Remove/Add/Change Default interface
 - Write Invoice Preference Get/Set interface
 - Write Account Overview interface
 - Implement Current Usage interface
 - Implement login with Basic authentication
 - Test Service Summary and List Services bugs
 - Write Password change interface
 - Add appropriate information to API call headers
 - Add Call ID to calls
 - Implement Error Checking and reporting
- Documentation
 - Write Error Documentation
 - Write API Documentation

Goals Met

This section will detail the goals we were able to meet for this sprint.

- Android
 - Technician Application
 - Set up saved Account functionality
 - Add a settings page
 - Redesign layout of Diagnostic data
 - Enable screen rotation
 - Enable horizontal scrolling
 - Handling UI while loading data
 - eBill Application
 - Redesign internal storage structure
 - Deploy both apps to Mitchell
- Apple
 - Technician Application
 - Implemented Save/Manage Accounts functionality

- Added Settings page
 - Implemented Log
 - Implemented Expandable Diagnostics List
 - Added Screen Rotation functionality
 - Implemented Initial URL Set up and Storage (Changeable in Settings)
- eBill Application
 - Implemented Sign in page
 - App-life data stored in internal hierarchical structure
 - Created database
 - Implemented Make Payments
 - Implemented display of real Current Usage data
 - Implemented Account real data display/editing capabilities
 - Added Invoice overview for past invoices
- API
 - Technician Application
 - Investigated IMSI or Device-Specific Identifiers for Authentication
 - Implemented Error Checking and reporting
 - Parsed Dates correctly
 - Added Call ID to calls
 - eBill Application
 - Tested making a new Payment, API not working yet
 - Parsed Dates correctly
 - Trouble Tickets were removed as a feature
 - Wrote Account List interface
 - Wrote Email Get/Remove/Add/Change Default interface
 - Wrote Invoice Preference Get/Set interface
 - Wrote Account Overview interface
 - Implemented Current Usage interface
 - Implemented login with Basic authentication
 - Tested Service Summary and List Services bugs (now working)
 - Password change was removed as a feature
 - Added appropriate information to API call headers
 - Added Call ID to calls
 - Implemented Error Checking and reporting
 - Documentation
 - Wrote Error Documentation

Goals Not Met

This section will detail the goals we did not meet for this sprint

- Android
 - eBill Application
 - Add database to project
 - Make Payments from phone
 - Retrieve real data for current usage
 - Read and edit real account settings data
- Apple
 - Technician Application
 - Horizontal Scrolling
 - eBill Application
 - Storing data in the database
- API
 - Technician Application
 - Write Unit Tests
 - Add Logging functionality
 - eBill Application
 - Write Unit Tests
 - Add Logging functionality
 - Documentation
 - Write API Documentation

Research/Code Experiments

Jeremy did research into available device identifiers for both cellular devices and non-cellular devices. This research resulted in us deciding to not use a device identifier for the Technician App.

Rachel and Mike did research into getting horizontal scrolling working for the Technician app.

Prototypes & Features

At the culmination of this sprint, we have a functional Technician App on both platforms, as well as an eBill app that is mostly functional. We are missing the final log in functionality on the Technician side, as our sponsors have not defined it yet. Some small features are missing on the eBill applications as we were unable to get all of the desired functionality implemented.

Product Backlog

- Non-App Specific
 - Handle navigation away from and back to the application
 - Implement Tablet Layout
 - Unit Tests
- Android
 - Technician Application
 - Implement Icons
 - 2-Pane mode for Tablets
 - eBill Application
- Apple
 - Technician Application
 - Implement Icons
 - 2-Pane mode for Tablets
 - Make application iPad compatible
 - eBill Application
 - Make application iPad compatible
- API
 - Add Date formatting options
 - Refactor Android calls
- Documentation
 - Work on CONOPS
 - Work on Requirements
 - Work on Design Document